

Visual Database Tools

Frank Pellow

fpellow@microsoft.com

Program Manager

Microsoft Corporation

A large space shuttle is shown launching vertically on the right side of the image. It has a white body with orange and black stripes. Bright orange and yellow flames and white smoke are coming from the engines at the bottom. In the top right corner, there are several small icons of computer windows or documents connected by lines.

POWER

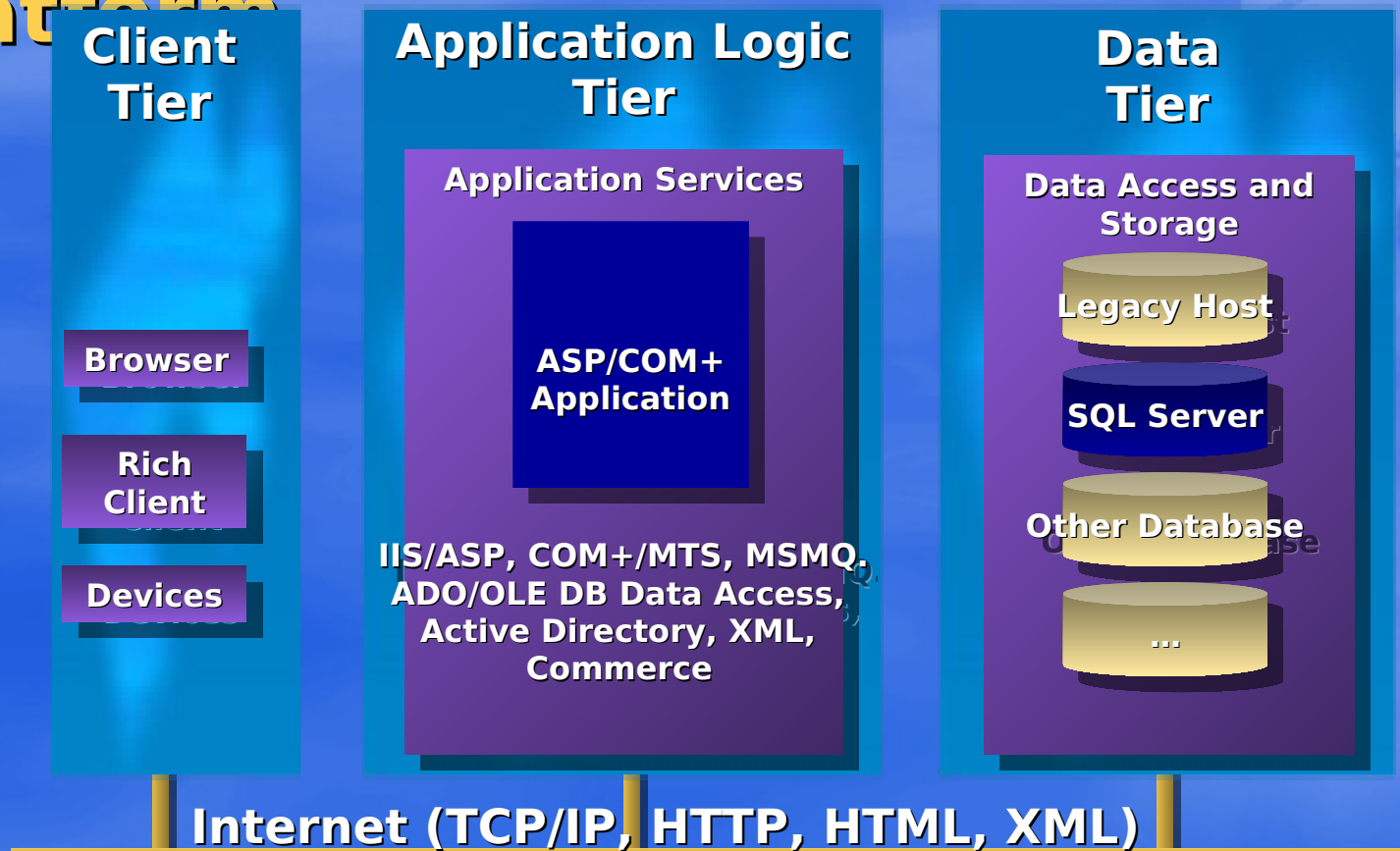
Windows DNA 2000

Readiness Conference

/// featuring SQL Server 2000

Windows DNA 2000

Next Generation Web Application Platform



Microsoft
SQL Server 2000
Server2000

Microsoft
Application Center 2000



Microsoft
Commerce Server 2000

Microsoft
Host Integration Server 2000

Microsoft
BizTalk Server 2000

Visual

- **Graphical Designers**

Database

- **SQL Server™, Jet, Fox, Oracle, etc.**

Tools

- **Database Designer**
- **Query Design Suite**
- **Database Explorer**
- **SQL Script Generator**
- **SQL Script Editor**
- **SQL Routine Debugger**

Microsoft
Office



Visual Database Tools



Microsoft
SQL Server



Oracle

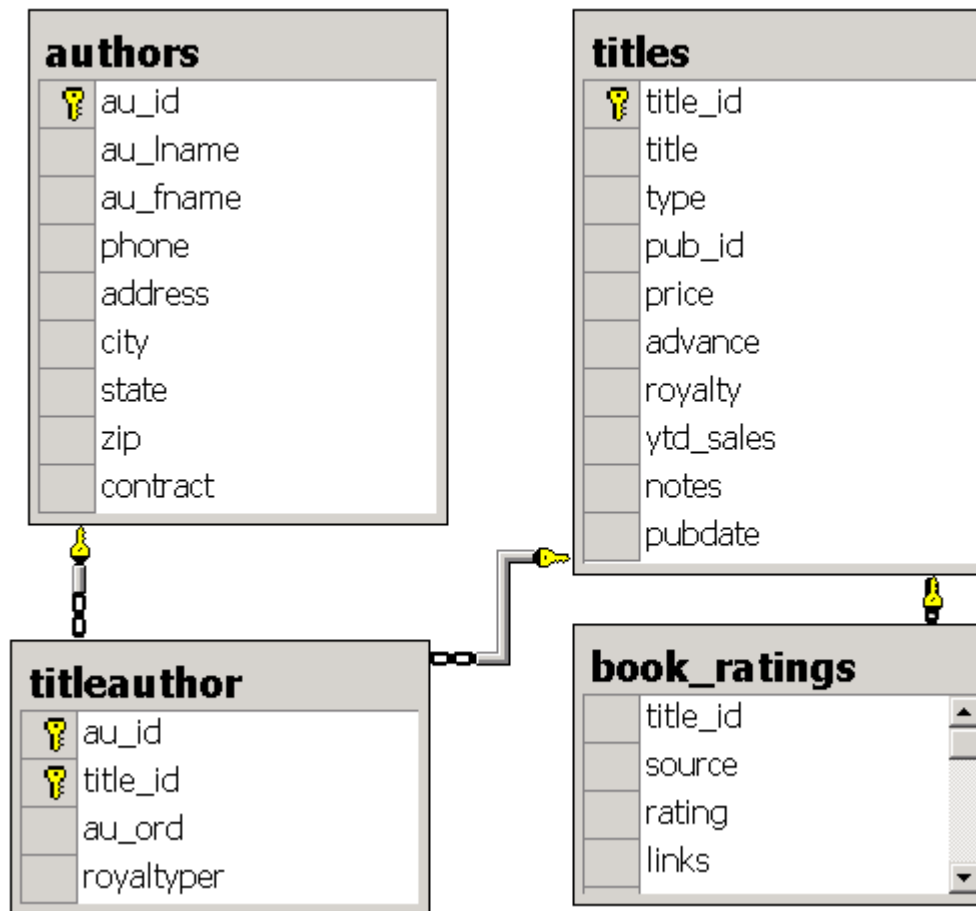
Other

Microsoft
**Visual
Studio**

Ubiquitous

- **Tools available in:**
 - **SQL Server Enterprise Manager**
 - (As of SQL Server 7.0)
 - **Access**
 - (As of Office 2000)
 - **Visual Studio®**
 - (As of Visual Studio 97 Enterprise Edition)

Visual Database Tools



**Database
Designer**

Visual Database Tools

group-by icon

Query Design Suite

diagram pane

grid pane

SQL pane

result pane

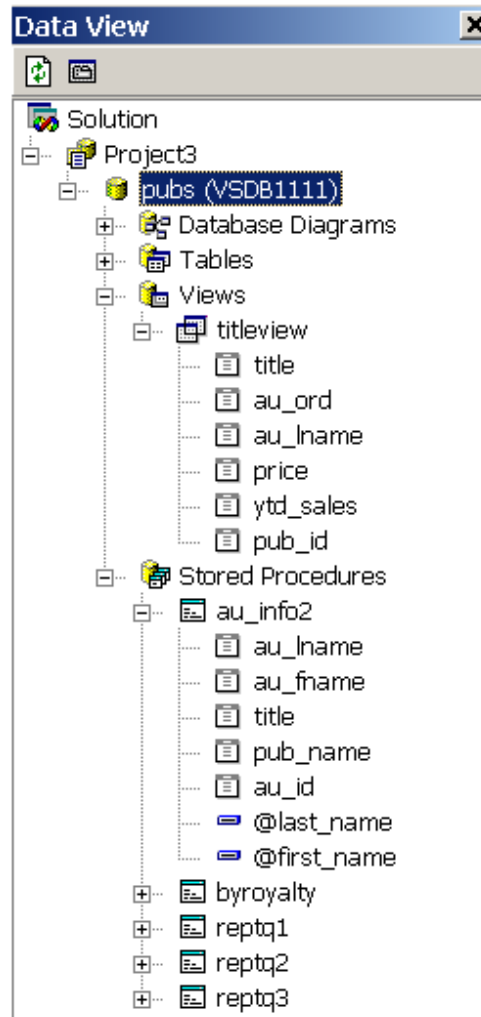
The screenshot shows a Visual Database Tool interface with four main panes. The top pane is the 'diagram pane' showing a query design diagram with two tables, 'publishers' and 'authors', connected by a 'join' line. The 'publishers' table has fields 'pub_id', 'pub_name', 'city', 'state', and 'country'. The 'authors' table has fields 'au_fname', 'phone', 'address', 'city', and 'state'. The 'grid pane' shows a table with columns: Column, Alias, Table, Output, Sort Type, Sort Order, Criteria, and Or... The 'SQL pane' shows the SQL query:

```
SELECT dbo.publishers.pub_name, dbo.publishers.state AS pub_state, dbo.publishers.city AS pub_city, dbo.authors.au_fname, dbo.authors.state AS author_state, dbo.authors.city AS author_city
FROM dbo.authors LEFT OUTER JOIN
    dbo.publishers ON dbo.authors.state = dbo.publishers.state
ORDER BY dbo.authors.state
```

 The 'result pane' shows the query results in a table with columns: pub_name, pub_state, pub_city, au_fname, au_fname, author_state, and auth... The results are as follows:

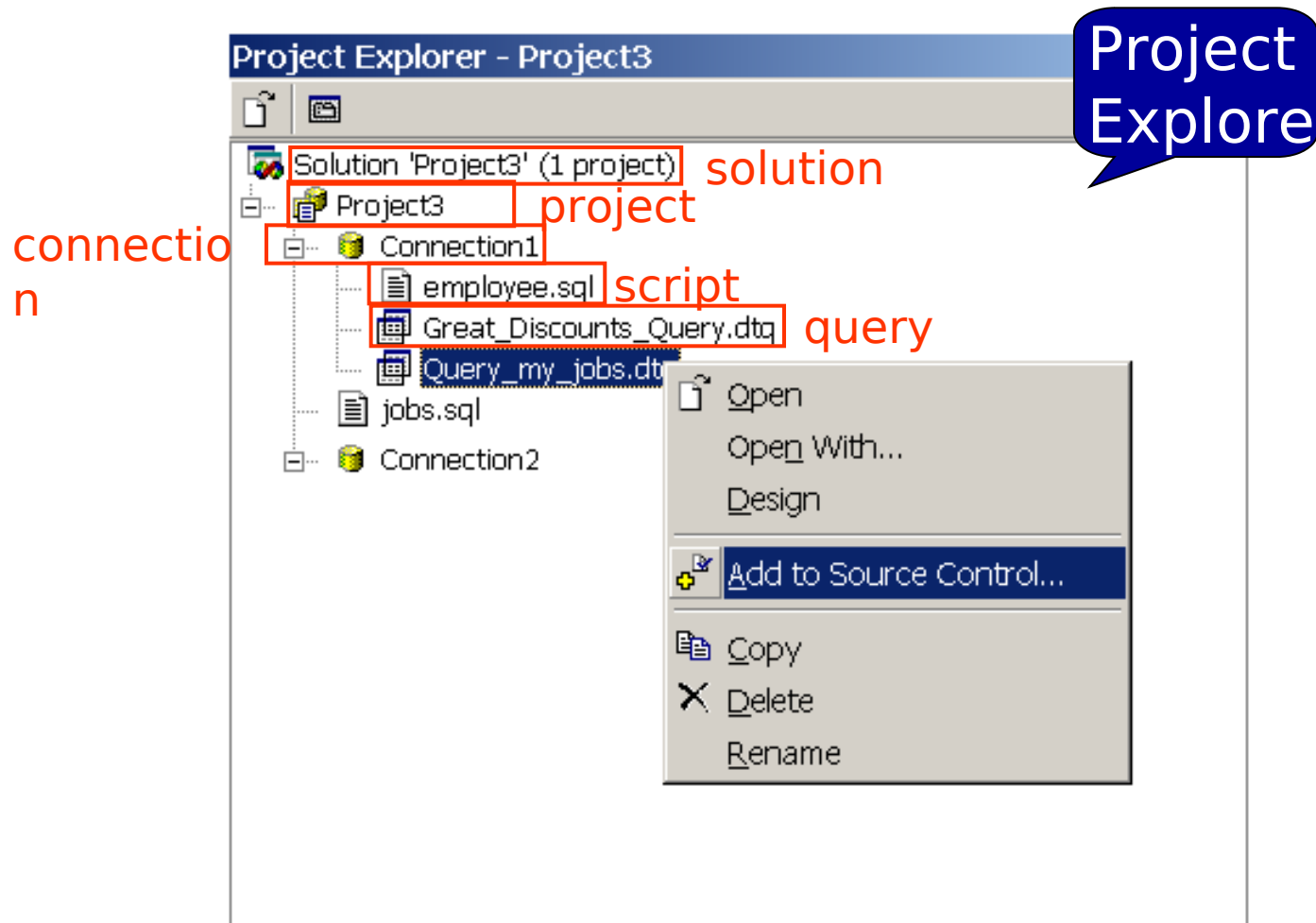
pub_name	pub_state	pub_city	au_fname	au_fname	author_state	auth...
Algodata Infosystem	CA	Berkeley	Hunter	Sheryl	CA	Palo
Algodata Infosystem	CA	Berkeley	McBadden	Heather	CA	Vaca
<NULL>	<NULL>	<NULL>	DeFrance	Michel	IN	Gary
<NULL>	<NULL>	<NULL>	Smith	Meander	KS	Lawr

Visual Database Tools



Database
e
Explorer

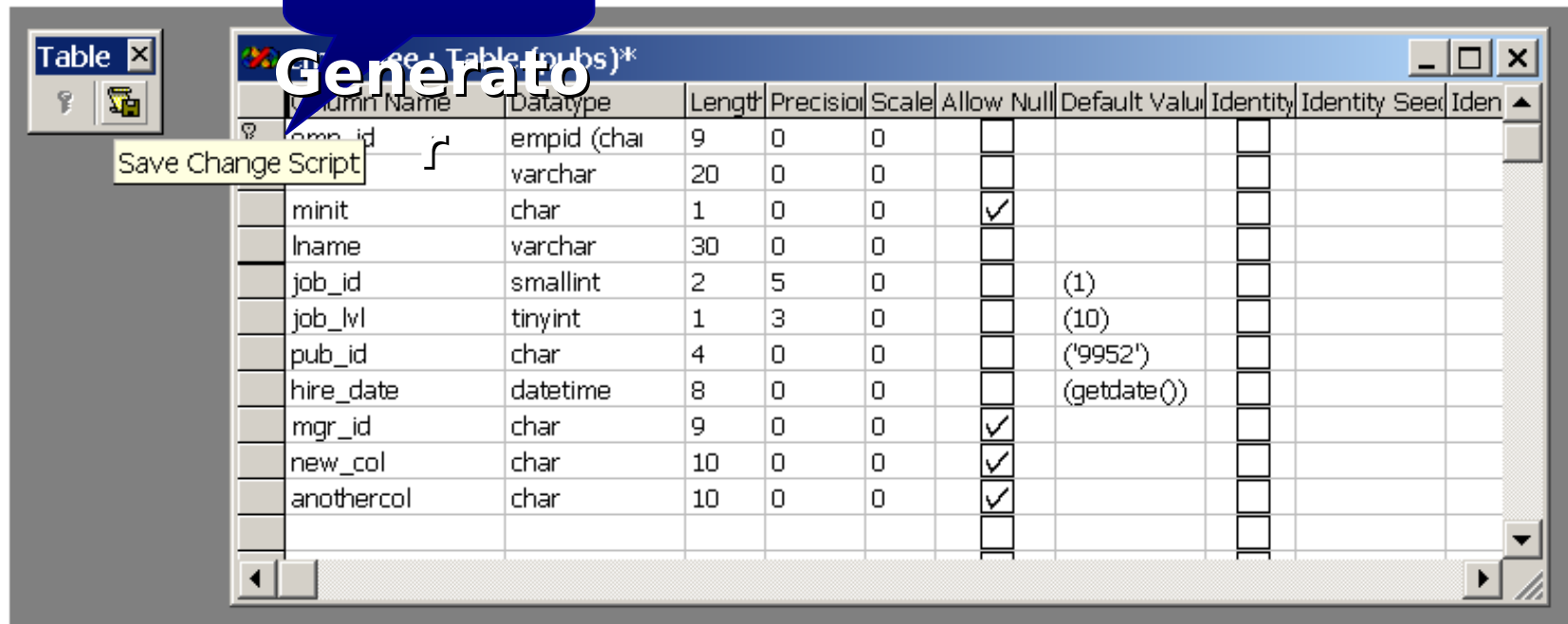
Visual Database Tools



Visual Database Tools

Script

Generator

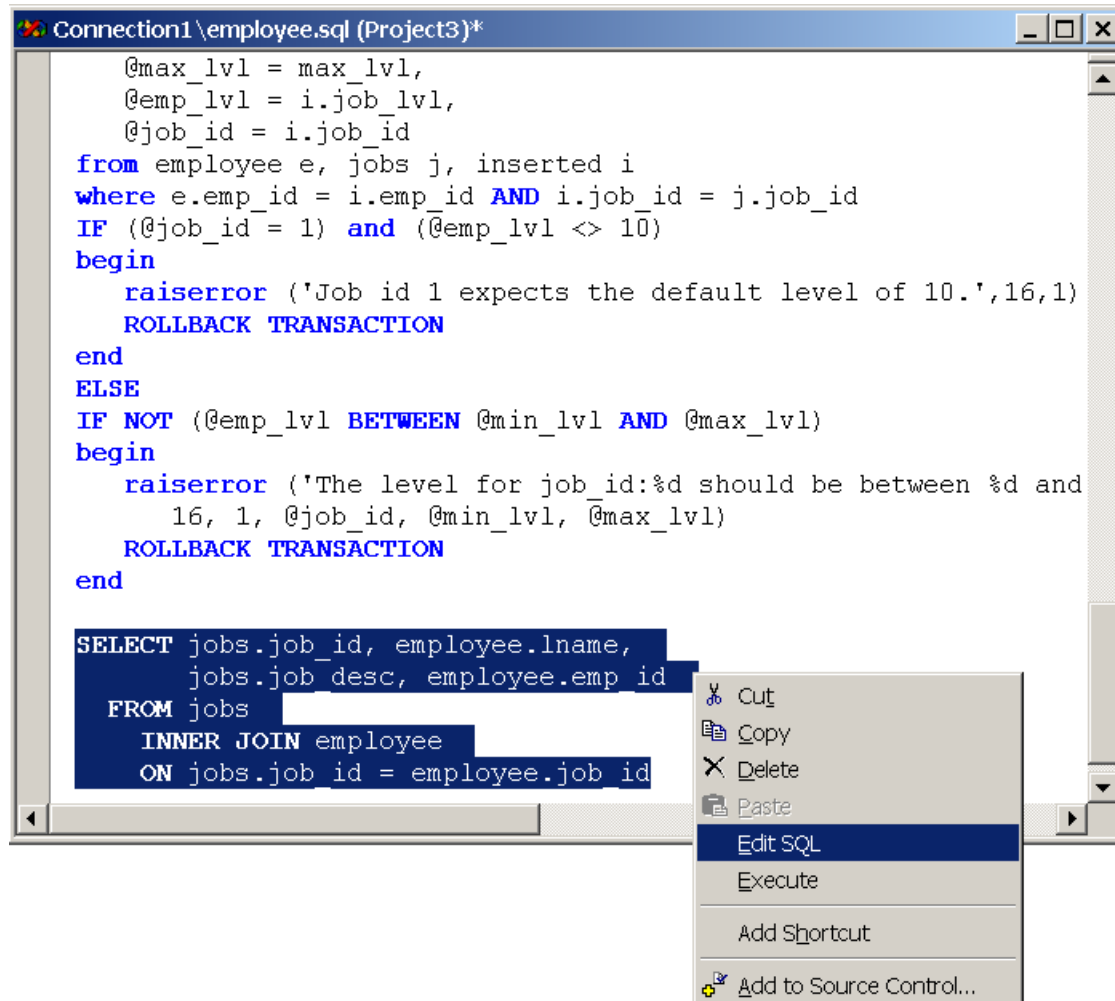


Table

Save Change Script

Column Name	Datatype	Length	Precision	Scale	Allow Null	Default Value	Identity	Identity Seed	Identity
emp_id	empid (char	9	0	0	<input type="checkbox"/>		<input type="checkbox"/>		
emp_name	varchar	20	0	0	<input type="checkbox"/>		<input type="checkbox"/>		
minit	char	1	0	0	<input checked="" type="checkbox"/>		<input type="checkbox"/>		
lname	varchar	30	0	0	<input type="checkbox"/>		<input type="checkbox"/>		
job_id	smallint	2	5	0	<input type="checkbox"/>	(1)	<input type="checkbox"/>		
job_lvl	tinyint	1	3	0	<input type="checkbox"/>	(10)	<input type="checkbox"/>		
pub_id	char	4	0	0	<input type="checkbox"/>	('9952')	<input type="checkbox"/>		
hire_date	datetime	8	0	0	<input type="checkbox"/>	(getdate())	<input type="checkbox"/>		
mgr_id	char	9	0	0	<input checked="" type="checkbox"/>		<input type="checkbox"/>		
new_col	char	10	0	0	<input checked="" type="checkbox"/>		<input type="checkbox"/>		
anothercol	char	10	0	0	<input checked="" type="checkbox"/>		<input type="checkbox"/>		

Visual Database Tools



```
Connection1\employee.sql (Project3)*

    @max_lvl = max_lvl,
    @emp_lvl = i.job_lvl,
    @job_id = i.job_id
from employee e, jobs j, inserted i
where e.emp_id = i.emp_id AND i.job_id = j.job_id
IF (@job_id = 1) and (@emp_lvl <> 10)
begin
    raiserror ('Job id 1 expects the default level of 10.',16,1)
    ROLLBACK TRANSACTION
end
ELSE
IF NOT (@emp_lvl BETWEEN @min_lvl AND @max_lvl)
begin
    raiserror ('The level for job_id:%d should be between %d and
    16, 1, @job_id, @min_lvl, @max_lvl)
    ROLLBACK TRANSACTION
end

SELECT jobs.job_id, employee.lname,
       jobs.job_desc, employee.emp_id
FROM jobs
     INNER JOIN employee
     ON jobs.job_id = employee.job_id
```

Context menu options:

- Cut
- Copy
- Delete
- Paste
- Edit SQL**
- Execute
- Add Shortcut
- Add to Source Control...

Script
Editor

Visual Database Tools

Connection1\employee.sql (Project3)*

```
@max_lvl = max_lvl
@emp_lvl = i.job_lvl
@job_id = i.job_id
from employee e, jobs
where e.emp_id = i.emp_id
IF (@job_id = 1) and
begin
    raiserror ('Job id', 16, 1, @job_id,
    ROLLBACK TRANSACTION
end
ELSE
IF NOT (@emp_lvl BETWEEN
begin
    raiserror ('The level', 16, 1, @emp_lvl,
    ROLLBACK TRANSACTION
end

SELECT jobs.job_id, employee.lname, jobs.job_desc,
       jobs.job_id, employee.emp_id
FROM jobs INNER JOIN employee
```

Query Builder : employee.sql

jobs

- ☐ * (All Columns)
- ☐ IDENTITYCOL
- ☒ job_id
- ☒ job_desc
- ☐ min_lvl

employee

- ☐ * (All Columns)
- ☒ emp_id
- ☐ fname
- ☐ minit
- ☒ lname

Column	Alias	Table	Output	Sort Type
job_id		jobs	<input checked="" type="checkbox"/>	
lname		employee	<input checked="" type="checkbox"/>	
inh desc		inhs	<input checked="" type="checkbox"/>	

SELECT jobs.job_id, employee.lname, jobs.job_desc,
employee.emp_id
FROM jobs INNER JOIN

Solution

- Project3
 - pubs (VSD81111)
 - Database Diagrams
 - DatabaseDiagram1
 - Tables
 - authors
 - book_ratings
 - discounts
 - employee
 - jobs
 - pub_info
 - publishers
 - roysched
 - sales
 - stores
 - TABLE1
 - titleauthor
 - titles
 - Views
 - Stored Procedures

Script

DML Designer

Visual Database Tools

Routine Debugger

star

insert breakpoint

watch region

The screenshot shows a SQL IDE window with a toolbar at the top. A red box highlights the 'star' icon (breakpoint) on the left. Another red box highlights the 'insert breakpoint' icon (hand) on the toolbar. A third red box highlights the 'watch region' icon (magnifying glass) on the toolbar. The main text area contains SQL code. A red box highlights the 'current position' icon (yellow arrow) on the left. A red box highlights the 'breakpoint' icon (red circle) on the left. A context menu is open over the code, showing options: Cut, Copy, Delete, Paste, Insert SQL, Debug, Insert Breakpoint, Add Watch, Run To Cursor, and Add Shortcut.

```
as
--create the temporary table that accumulated result:
create table #Reports_Holding_Table (EmpId char(9) primary key,
                                     EmpName varchar(51) NOT NULL,
                                     MgrId char(9),
                                     JobLvl tinyint,
                                     JobId smallint,
                                     Processed tinyint default 0)

-- variable to hold number of rows added
declare @RowsAdded int

--Initialize #Reports_Holding_Table with direct reports of the given empl
insert into #Reports_Holding_Table
select emp_id, fname+' '+lname, mgr_id, job_lvl, job_id, 0
from employee
where mgr_id = @InEmpId |
set @RowsAdded = @@rowcount

--While new employees were added in the previous ite
while @RowsAdded > 0
begin
    -- Mark all employee records whose direct report
    -- found in this iteration.
    update #Reports_Holding_Table
    set processed = 1
    where processed = 0

    -- Insert employees who report to employees mark
    insert into #Reports_Holding_Table
```

current
position

break
point

In Visual Studio 6.0

Generic

Visually design databases

Visually design queries

Visually design views

Explore databases

Generate deployment scripts

Create/edit procedures/triggers

Debug procedures/triggers

SQL
Server

Oracle



Visual Studio 6.0 Demo

(with SQL Server 7.0)

Data View Explorer & Project Explorer

Database Designer

Query Designer

Stored Procedure Debugger

In Access 2000

- **MSDE 1.0 (SQL run-time)** is an “industrial strength” alternative to Jet
- First step to a Jet-like GUI for SQL Server
- Visually design databases
- Visually design views
- Visually design queries that supply values for forms

Access 2000 Demo

SQL Server 7.0 via an Access Project

DML Designer with an Access form

In SQL Server 7.0

- Visually design databases
- Visually design views
- Visually design: **SELECT**
INSERT
UPDATE
DELETE
MAKE TABLE

In SQL Server 2000

- **Keeping pace with SQL Server 2000**
 - **SQL_VARIANT and BIGINT**
 - **Indexed views**
 - **Cascade referential Integrity**
 - **User-defined functions**
 - **Extended properties**
- **Independent improvements**
 - **Better table designer**
 - **Property page additions**

SQL Server 2000 Demo

Database Designer with new SQL

View Designer with an indexed view

**DML Designer with a table-valued
function**

**DML Designer transforming
statement types**

Summary

- Visual Tools well integrated into 3 hosts
- Making SQL much easier to use
- Keeping current with (and even driving) SQL Server enhancements
- Keeping current with Access and Visual Studio enhancements
- Always improving and extending the tools

What Next?

- **Use and comment upon our tools in SQL Server Beta 2**
- **Use our tools in Visual Studio 6.0 & Access 2000**
- **Apply for Visual Studio 7.0 Beta 1**
- **Watch for Office 10 beta program**

Microsoft
Office



Visual Database Tools



Microsoft
SQL Server



Oracle

Other

Microsoft
**Visual
Studio**

POWER

UP



Microsoft®